|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
|  |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| Институт комплексной безопасности и специального приборостроения |
| Кафедра КБ-2 «Прикладные информационные технологии» |

**Лабораторная работа по Python**

по дисциплине: **«Технологии и методы программирования»**.

Подготовил студент 3 курса группы БИСО-01-20 Полянская Полина Алексеевна

Студенческий: 20Б0866

Проверил преподаватель Выжигин Александр Юрьевич.

Содержание

**Общее задание** **3**

**Номер варианта, текст варианта** **4**

**Описание логической структуры программы5**

**Скриншоты работы программы8**

Общее задание

Размерности массивов следует задать именованными константами. Все необходимые данные должны передаваться подпрограммам в качестве параметров; все величины, используемые только внутри подпрограмм, должны быть описаны как локальные. Использование глобальных переменных в подпрограммах не допускается. Вывод результатов работы подпрограмм должен выполняться в главной программе.

Номер варианта, текст варианта

Вариант 9

Соседями элемента Aij в матрице назовем элементы Аkl, где і - 1 <= k <= i + 1,

j - 1 <= l <= j + 1, (k, l) != (i, j). Операция сглаживания матрицы дает новую матрицу того же размера, каждый элемент которой получается как среднее арифметическое имеющихся соседей соответствующего элемента исходной матрицы. Построить результат сглаживания заданной вещественной матрицы размером 10 × 10 (оформить в виде процедуры).

В сглаженной матрице найти сумму модулей элементов, расположенных ниже главной диагонали (оформить в виде функции).

Описание логической структуры программы

from random import uniform

n = int(input("сколько строк в матрице? "))

m = int(input("сколько столбцов в матрице? "))

A = []

for i in range(n):

A.append( [0]\*m )

S = []

for i in range(n):

S.append( [0]\*m )

f = int(input("Введите, если: массив наполнить рандомно 0, с клавиатуры 1 "))

if f:

for i in range(n):

for j in range(m):

A[i][j] = float(input(f'введите элемент:A[{i}][{j}]='))

else:

for i in range(n):

for j in range(m):

A[i][j] = uniform(1,30)

def sglazhivanie(A,S,n,m):

for i in range(n):

for j in range(m):

s = 0;

kolv = 0

for k in range(i-1,i+2):

for l in range(j-1,j+2):

if ((k,l) != (i,j)) and (k>=0) and (l>=0) and (k<n) and (l<m):

s = s + A[k][l]

kolv = kolv + 1

if kolv>0:

s = s / kolv

S[i][j] = s

sglazhivanie(A,S,n,m)

def sum\_abs\_niz\_gd(S, n, m):

summ = 0

for i in range(n):

for j in range(m):

if i>j:

summ = summ + abs(S[i][j])

return summ

print("A:")

for i in range(n):

for j in range(m):

print(A[i][j], end=" ")

print()

print("S:")

for i in range(n):

for j in range(m):

print(S[i][j], end=" ")

print()

print()

print(f"Сумма модулей элементов, расположенных ниже главной диагонали = {sum\_abs\_niz\_gd(S,n,m)}")

Скриншоты работы программы

